

10/19/00
JC904 U.S. PTO

10-20-00

EK538216795US

A

THE ASSISTANT COMMISSIONER FOR PATENTS
Washington, D.C. 20231

DOCKET NUMBER: **AUS920000621US1**
DATE: October 19, 2000

JC759 U.S. PTO
09/692350
10/19/00

Sir:

Transmitted herewith for filing is the Patent Application of:

Inventors: **Renato John Recio et al**

For: **EFFICIENT PROCESS FOR HANDOVER BETWEEN SUBNET MANAGERS**

Enclosed are:

☒ Patent Specification and Declaration (3 Declarations)

10 sheets of drawing(s). (Informal)

☒ An assignment of the invention to International Business Machines Corporation (includes Recordation Form Cover Sheet).

☐ A certified copy of a ☐ application.

☐ Information Disclosure Statement, PTO 1449 and copies of references.

The filing fee has been calculated as shown below:

For	Number Filed	Number Extra	Rate	Fee
Basic Fee				\$710.00
Total Claims	20 - 20	0	x 18 =	\$ 0.00
Indep. Claims	3 - 3	0	x 80 =	\$ 0.00
MULTIPLE DEPENDENT CLAIM PRESENTED			x 270 =	\$ 0.00
TOTAL				\$710.00

☒ Please charge my Deposit Account No. 09-0447 in the amount of \$ 710.00. A duplicate copy of this sheet is enclosed.

☒ The Commissioner is hereby authorized to charge payment of the following fees associated with this communication or credit any overpayment to Deposit Account 09-0447. A duplicate copy of this sheet is enclosed.

☒ Any additional filing fees required under 37 CFR section 1.16.

☒ Any patent application processing fees under 37 CFR section 1.17.

Respectfully submitted,

By Mark E. McBurney
Mark E. McBurney
Registration No. 33,114
Intellectual Property Law Dept.
IBM Corporation
11400 Burnet Road, Zip 4054
Austin, Texas 78758
Telephone (512) 823-1003

EFFICIENT PROCESS FOR HANDOVER BETWEEN SUBNET MANAGERS

5 The present invention is related to the subject
matter of the following commonly assigned, co-pending
United States Patent Applications filed concurrently
herewith: Serial No. _____ (Docket No. AUS9-2000-0620)
entitled "Method and System for Informing An Operating
System In A System Area Network When A New Device Is
10 Connected"; Serial No. _____ (Docket No. AUS9-2000-
0622) entitled "Method and System For Scalably Selecting
Unique Transaction Identifiers"; Serial No. _____
(Docket No. AUS9-2000-0623) entitled "Method And System
For Reliably Defining and Determining Timeout Values In
15 Unreliable Datagrams"; and Serial No. _____ (Docket No.
AUS9-2000-0624) entitled "Method and System For Choosing
A Queue Protection Key That is Tamper-proof From An
Application". The content of the above-referenced
applications is incorporated herein by reference.

BACKGROUND OF THE INVENTION**1. Technical Field:**

25 The present invention relates in general to computer
networks and, in particular, to merging of independent
computer networks. Still more particularly, the present
invention relates to a method and system for providing an
30 efficient handover of control between Subnet managers of
separate subnets, which are being merged into a single
subnet.

2. Description of the Related Art:

The use of I/O interconnects to connect components of a distributed computer system is known in the art. Traditionally, in such systems, individual components are interconnected via a parallel bus, such as a PCI bus. The parallel bus has a relatively small number of plug-in ports for connecting the components. The number of plug-in-ports is set (i.e., the number cannot be increased). At maximum loading, a PCI bus transmits data at about 1 Gbyte/second.

The introduction of high performance adapters (e.g., SCSI adapters), Internet-based networks, and other high performance network components has resulted in increased demand for bandwidth, faster network connections, distributed processing functionality, and scaling with processor performance. These and other demands are quickly outpacing the current parallel bus technology and are making the limitations of parallel buses even more visible. PCI bus, for example, is not scalable, i.e., the length of the bus and number of slots available at a given frequency cannot be expanded to meet the needs for more components, and the limitation hinders further development of fast, efficient distributed networks, such as system area networks. New switched network topologies and systems capable of being easily expanded are required to keep up with the increasing demands, while allowing the network processes on the expanding network to be dynamically completed. i.e., without manual input.

The present invention recognizes the need for

faster, more efficient computer interconnects offering the features demanded by the developments of technology. More specifically, the present invention recognizes the need for providing a mechanism within a network such as a System Area Network (SAN) consisting of multiple subnets, that provides efficient, dynamic combining of two or more subnets into a single network.

SUMMARY OF THE INVENTION

5 A method and system for efficiently combining (or merging) subnets having individual master subnet managers into a single network with one master subnet manager is disclosed. The invention is thus applicable to a distributed computing system, such as a system area network, having end nodes, switches, and routers, and links interconnecting these components. The switches and routers interconnect the end nodes and route packets, i.e., sub-components of messages being transmitted, from a source end node to a target end node. The target end node then reassembles the packets into the message.

10
15 During discovery and configuration of a subnet, a subnet manager creates a Subnet Management Database (SMDB) representative of the subnet components being managed. Each subnet manager contains an independent SMDB. When two or more subnets are merged (i.e., linked/connected together) to form a single network, a single one of the subnet managers is selected as the master subnet manager and all the subnets SMDBs must be merged. The other subnet managers are relegated to standby status. In a preferred embodiment, a SMDB record labeling mechanism is utilized to differentiate among components from the different subnets that may have the same parameter values, such as protection keys (P_keys).

20
25
30 In one embodiment of the invention, when there are two or more separate SMDBs, each maintained by separate master subnet managers on a separate subnet, when those separate subnets are linked together requiring they be merged into

5

All objects, features, and advantages of the present invention will become apparent in the following detailed written description.

BRIEF DESCRIPTION OF THE DRAWINGS

The novel features believed characteristic of the invention are set forth in the appended claims. The invention itself however, as well as a preferred mode of use, further objects and advantages thereof, will best be understood by reference to the following detailed description of an illustrative embodiment when read in conjunction with the accompanying drawings, wherein:

Figure 1A depicts a system area network (SAN) in which the present invention is preferably implemented;

Figure 1B is a diagram illustrating the software aspects of SAN management model in accordance with the present invention;

Figure 2 is a diagram illustrating software aspects of an exemplary host processor end node for the SAN of **Figure 1** in accordance with the present invention;

Figure 3 is a diagram of an exemplary host channel adapter of the SAN of **Figure 1** in accordance with the present invention;

Figure 4 is a diagram illustrating processing of work requests with queues in accordance with a preferred embodiment of the present invention;

Figure 5 is an illustration of a data packet in accordance with a preferred embodiment of the present invention;

Figure 6 is a diagram illustrating a communication over a portion of a SAN fabric;

Figure 7 is a diagram illustrating packet transfers in accordance with the invention;

Figure 8 is a diagram illustrating two independent networks, which are linked together to form a single network in accordance with a preferred embodiment of the invention;

Figure 9A is a flow chart showing the process of time stamping SMDB entries in accordance with a preferred embodiment of the invention; and

Figure 9B is a flow chart showing the process absorbing an SMDB by a master subnet manager in accordance a preferred embodiment of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

5 The present invention is directed to a method for efficient handover of control to a single master subnet manager when two or more subnets having subnet managers are merged. The invention is applicable to a distributed computer system, such as a system area network (SAN). The invention is implemented in a manner that allows the merged subnet overlaps in parameters, such as P_Keys, to be resolved.

10 In order to appreciate the environment within which the invention is preferably practiced, a description of a SAN configured with routers, switches, and end nodes, etc. is provided below. Presentation of the environment and particular functional aspects of the environment which enable the invention to be practiced are provided with reference to **Figures 1-5**. Section headings have been provided to distinguish the hardware and software architecture of the SAN. However, those skilled in the art understand that the descriptions of either architecture necessarily includes references to both components.

25 SAN HARDWARE ARCHITECTURE

30 With reference now to the figures and in particular with reference to **Figure 1**, there is illustrated an exemplary embodiment of a distributed computer system. Distributed computer system **100** represented in **Figure 1** is provided merely for illustrative purposes, and the embodiments of the present invention described below can

be implemented on computer systems of numerous other types and configurations. For example, computer systems implementing the present invention may range from a small server with one processor and a few input/output (I/O) adapters to very large parallel supercomputer systems with hundreds or thousands of processors and thousands of I/O adapters. Furthermore, the present invention can be implemented in an infrastructure of remote computer systems connected by an Internet or intranet.

As shown in **Figure 1**, distributed computer system **100** includes a system area network (SAN) **113**, which is a high-bandwidth, low-latency network interconnecting nodes within the distributed computer system. More than one (1) SAN **113** may be included in a distributed computer system **100** and each SAN **113** may comprise multiple sub-networks (subnets).

A node is herein defined to be any component that is attached to one or more links of a network. In the illustrated distributed computer system, nodes include host processors **101**, redundant array of independent disks (RAID) subsystem **103**, I/O adapters **105**, switches **109A-109C**, and router **111**. The nodes illustrated in **Figure 1** are for illustrative purposes only, as SAN **113** can connect any number and any type of independent nodes. Any one of the nodes can function as an end node, which is herein defined to be a device that originates or finally consumes messages or frames in the distributed computer system **100**.

SAN **113** is the communications and management infrastructure supporting both I/O and inter-processor communications (IPC) within distributed computer system **100**. Distributed computer system **100**, illustrated in **Figure 1**, includes a switched communications fabric (i.e., links, switches and routers) allowing many devices to concurrently transfer data with high-bandwidth and low latency in a secure, remotely managed environment. End nodes can communicate over multiple ports and utilize multiple paths through SAN **113**. The availability of multiple ports and paths through SAN **113** can be employed for fault tolerance and increased-bandwidth data transfers.

SAN **113** includes switches **109A-109C** and routers **111**. Switch **109A-109C** connects multiple links together and allows routing of packets from one link to another link within SAN **113** using a small header Destination Local Identifier (DLID) field. Router **111** is capable of routing frames from one link in a first subnet to another link in a second subnet using a large header Destination Globally Unique Identifier (DGUID). Router **111** may be coupled via wide area network (WAN) and/or local area network (LAN) connections to other hosts or other routers.

In SAN **113**, host processor nodes **101** and I/O nodes **106** include at least one Channel Adapter (CA) to interface to SAN **113**. Host processor nodes **101** include central processing units (CPUs) **119** and memory **121**. In one embodiment, each CA is an endpoint that implements

the CA interface in sufficient detail to source or sink packets transmitted on SAN **113**. As illustrated, there are two CA types, Host CA (HCA) **117** and Target CA (TCA) **127**. HCA **117** is used by general purpose computing nodes to access SAN **113**. In one implementation, HCA **117** is implemented in hardware. In the hardware implementation of HCA **117**, HCA hardware offloads much of CPU and I/O adapter communication overhead. The hardware implementation of HCA **117** also permits multiple concurrent communications over a switched network without the traditional overhead associated with communicating protocols. Use of HCAs **117** in SAN **113** also provides the I/O and IPC consumers of distributed computer system **100** with zero processor-copy data transfers without involving the operating system kernel process. HCA **117** and other hardware of SAN **113** provide reliable, fault tolerant communications.

The I/O chassis **106** includes I/O adapter backplane and multiple I/O adapter nodes **105** that contain adapter cards. Exemplary adapter cards illustrated in **Figure 1** include SCSI adapter card **123A**, adapter card **123B** to fiber channel hub and FC-AL devices, Ethernet adapter card **123C**, graphics adapter card **123D**, and video adapter card **123E**. Any known type of adapter card can be implemented. The I/O chassis **106** also includes switch **109B** in the I/O adapter backplane to couple adapter cards **123A-123E** to SAN **113**.

RAID subsystem **103** includes a microprocessor **125**, memory **126**, a Target Channel Adapter (TCA) **127**, and

multiple redundant and/or striped storage disks **129**.

In the illustrated SAN **113**, each link **115** is a full duplex channel between any two network elements, such as end nodes, switches **109A-109C**, or routers **111**. Suitable links **115** may include, but are not limited to, copper cables, optical cables, and printed circuit copper traces on backplanes and printed circuit boards. The combination of links **115** and switches **109A-109C**, etc. operate to provide point-to-point communication between nodes of SAN **113**.

SAN SOFTWARE ARCHITECTURE

Software Components

Software and hardware aspects of an exemplary host processor node **101** are generally illustrated in **Figure 2**. Host processor node **101** includes one or more processors that execute a set of consumer processes **201**. Host processor node **101** includes HCA **117** with ports **205**. Each port **205** connects to a link **115** of SAN **113**. Ports **205** can connect to one SAN subnet or multiple SAN subnets. Utilizing message and data services **203**, consumer processes **201** transfer messages to SAN **113** via verbs interface **207**. Verbs interface **207** is generally implemented with an operating-system specific programming interface.

A software model of HCA **117** is illustrated in **Figure 3**. HCA **117** includes a set of queue pairs (QPs) **301**, which transfer messages across ports **205** to the subnet.

5 A single HCA **117** may support thousands of QPs **301**. By contrast, TCA **127** in an I/O adapter typically supports a much smaller number of QPs **301**. Also illustrated are subnet management administration (SMA) **209**, management packets **211** and a number of virtual lanes **213**, which connect transport layer with ports **205**.

10 Turning now to **Figure 5**, there is illustrated a software management model for nodes on SAN **113**. SAN architecture management facilities provides a Subnet Manager (SM) **303A**, a Subnet Administration (SA) **303B**, and an infrastructure that supports a number of general management services. The management infrastructure includes a Subnet Management Agent (SMA) **307** operating in
15 each node and defines a general service interface that allows additional general services agents. Also, SAN architecture defines a common management datagram (MAD) message structure for communicating between managers and management agents.

20 SM **303A** is responsible for initializing, configuring and managing switches, routers, and channel adapters. The SM can be implemented within other devices, such as a channel adapter or a switch. One SM **303A** of SAN is
25 dedicated as a master SM and is responsible for:
discovering the subnet topology; configuring each channel adapter port with a range of Local Identification (LID) numbers, Global Identification (GID) number, subnet prefix, and Partition Keys (P_Keys); configuring each
30 switch with a LID, the subnet prefix, and with its forwarding database; and maintaining the end node and

service databases for the subnet to provide a Global Unique Identification (GUID) number to LID/GID resolution service as well as a services directory. Thus, management of SAN **113** and SAN components, such as HCAs **117**, TCAs (or end nodes) **127**, switches **109**, and routers **111** are completed utilizing Subnet Management (SM) **303A** and Subnet Administration (SA) **303B**. SMPs are used to discover, initialize, configure, and maintain SAN components through management agents **307** of end nodes **305**. SAN SA packets are used by SAN components to query and update subnet management data. Control of some aspects of the subnet management are provided via a user management console **311** in host-based end node **309**.

MESSAGE TRANSFER PROCESS

SAN **113** provides the high-bandwidth and scalability required for I/O and also supports the extremely low latency and low CPU overhead required for Interprocessor Communications (IPC). User processes can bypass the operating system (OS) kernel process and directly access network communication hardware, such as HCAs **117**, which enable efficient message passing protocols. SAN **113** is suited to current computing models and is a building block for new forms of I/O and computer cluster communication. SAN **113** allows I/O adapter nodes **105** to communicate among themselves or communicate with any or all of the processor nodes **101** in the distributed computer system. With an I/O adapter attached to SAN **113**, the resulting I/O adapter node **105** has substantially the same communication capability as any processor node

101 in the distributed computer system.

For reliable service types of messages, end nodes, such as host processor nodes **101** and I/O adapter nodes **105**, generate request packets and receive acknowledgment packets. Switches **109A-109C** and routers **111** pass packets along from the source to the target (or destination). Except for the variant CRC trailer field, which is updated at each transfer stage in the network, switches **109A-109C** pass the packets along unmodified. Routers **111** update the variant CRC trailer field and modify other fields in the header as the packet is routed.

In SAN **113**, the hardware provides a message passing mechanism that can be used for Input/Output (I/O) devices and Interprocess Communications (IPC) between general computing nodes. Consumers (i.e., processing devices connected to end nodes) access SAN **113** message passing hardware by posting send/receive messages to send/receive work queues (WQ), respectively, on a SAN Channel Adapter (CA).

A message is herein defined to be an application-defined unit of data exchange, which is a primitive unit of communication between cooperating processes. A packet (or frame) is herein defined to be one unit of data encapsulated by networking protocol headers (and trailer). The headers generally provide control and routing information for directing the packet (or frame) through SAN **113**. The trailer generally contains control and cyclic redundancy check (CRC) data for ensuring that frames are not delivered with corrupted content.

Consumers use SAN verbs to access HCA functions. The software that interprets verbs and directly accesses the CA is known as the Channel Interface (CI) **219**. Send/Receive work queues (WQ) are assigned to a consumer as a Queue Pair (QP). Messages may be sent over five different transport types, Reliable Connected (RC), Reliable Datagram (RD), Unreliable Connected (UC), Unreliable Datagram (UD), and Raw Datagram (RawD). Consumers retrieve the results of these messages from a Completion Queue (CQ) through SAN send and receive work completions (WC). The source CA takes care of segmenting outbound messages and sending them to the destination. The destination or target CA takes care of reassembling inbound messages and placing them in the memory space designated by the destination's consumer. These features are illustrated in the figures below.

Referring now to **Figure 4**, there is illustrated a block diagram of work and completion queue processing. Each QP **301** provides an input to a Send Work Queue (SWQ) **407** and a Receive Work Queue (RWQ) **409**. SWQ **407** sends channel and memory semantic messages, and RWQ **409** receives channel semantic messages. A consumer calls a verb (within verbs interface **207**) to place Work Requests (WRs) into a WQ. A Send WR **403** is a channel semantic operation to push a set of local data segments **417** to the data segments referenced by a remote node's Receive WQE **405**. Each of the Send WR's data segments **417** contains a virtually contiguous memory region. The virtual addresses used to reference the local data segments **417** are in the address context of the process that created

the local QP **301**.

As shown in **Figure 4**, WRs **403** that have been placed onto a WQ by consumer process **401** are referred to as work queue elements (WQEs) **405**. WQEs **405** are executed by hardware **415** in HCA **117**. SWQ **407** contains WQEs **405** that describe data to be transmitted on the SAN fabric. RWQ **409** contains WQEs **405** that describe where to place incoming channel semantic data received from SAN **113**.

In one embodiment, Receive Work Queues **409** only support one type of WQE **405**, which is referred to as a receive WQE. The receive WQE provides a channel semantic operation describing a local memory space into which incoming send messages are written. The receive WQE includes a scatter list describing several virtually contiguous memory spaces. An incoming send message is written to these memory spaces. The virtual addresses are in the address contexts of the process that created the local QP **301**.

Verbs interface **207** also provide a mechanism for retrieving completed work from completion queue **411**. Completion queue **411** contains Completion Queue Elements (CQEs) **413** which **413** contain information about previously completed WQEs **405**. Completion queues **413** are employed to create a single point of completion notification for multiple QPs **301**. CQE **413** contains sufficient information to determine the QP **301** and specific WQE **405** that completed. A completion queue context (not shown) is a block of information that contains pointers to,

length, and other information needed to manage individual completion queues **411**.

REMOTE OPERATION FUNCTIONALITY

5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
80
85
90
95
100
105
110
115
120
125
130
135
140
145
150
155
160
165
170
175
180
185
190
195
200
205
210
215
220
225
230
235
240
245
250
255
260
265
270
275
280
285
290
295
300
305
310
315
320
325
330
335
340
345
350
355
360
365
370
375
380
385
390
395
400
405
410
415
420
425
430
435
440
445
450
455
460
465
470
475
480
485
490
495
500
505
510
515
520
525
530
535
540
545
550
555
560
565
570
575
580
585
590
595
600
605
610
615
620
625
630
635
640
645
650
655
660
665
670
675
680
685
690
695
700
705
710
715
720
725
730
735
740
745
750
755
760
765
770
775
780
785
790
795
800
805
810
815
820
825
830
835
840
845
850
855
860
865
870
875
880
885
890
895
900
905
910
915
920
925
930
935
940
945
950
955
960
965
970
975
980
985
990
995

SAN **113**, with its interlinked arrangement of components and sub-components, provides a method for completing remote operations, by which processor nodes may directly control processes in I/O nodes. Remote operation also permits the network to manage itself. A remote direct memory access (RDMA) Read WR provides a memory semantic operation to read a virtually contiguous memory space on a remote node. A memory space can either be a portion of a memory region or a portion of a memory window. A memory region references a previously registered set of virtually contiguous memory addresses defined by a virtual address and length. A memory window references a set of virtually contiguous memory addresses which have been bound to a previously registered region.

The RDMA Read WR writes the data to a virtually contiguous local memory space. Similar to Send WR **403**, virtual addresses used by the RDMA Read WQE to reference the local data segments are in the address context of the process that created the local QP **301**. The remote virtual addresses are in the address context of the process owning the remote QP targeted by the RDMA Read WQE.

RDMA Write WQE provides a memory semantic operation to write a virtually contiguous memory space on a remote node. RDMA Write WQE contains a scatter list of local

virtually contiguous memory spaces and the virtual address of the remote memory space into which the data from the local memory spaces is written.

5 RDMA FetchOp WQE provides a memory semantic operation to perform an atomic operation on a remote word. RDMA FetchOp WQE is a combined RDMA Read, Modify, and Write operation. RDMA FetchOp WQE can support several read-modify-write operations, such as "Compare and Swap if Equal."

10 A Bind (unbind) remote access key (R_Key) WQE provides a command to the HCA hardware to modify a memory window by associating the memory window to a memory region. A second command to destroy a memory window by disassociating the memory window to a memory region is also provided. The R_Key is part of each RDMA access and is used to validate that the remote process has permitted access to the buffer.

20 EFFICIENT PROCESS FOR HANDOVER BETWEEN SUBNET MANAGERS

The present invention makes use of the features described in the subject matter of Patent Application Serial No. _____ (Attorney Docket No AUS9-2000-0625US1) "ASSOCIATION OF END-TO-END CONTEXT VIA RELIABLE DATAGRAM DOMAINS" filed on _____, the entire content of which is hereby incorporated by reference. The reference application allows Reliable Datagram QPs to be used for communicating across multiple partitions. In SAN 113, QPs that support SAN Service Types are associated to a partition and cannot communicate to QPs that are outside of the partition to which the QP is

associated. The QP is barred from communicating with QPs in another partition even if the node's HCA port, which the QP uses, has access to different partitions. RD QPs, however, can communicate with any given partition the node's HCA has access to, so long as there is an underlying End-End Context (EEC) which is associated with the given partition.

The present invention provides a method for seamlessly merging (i.e., linking or connecting) two independent subnets having separate subnet managers (SMs) into a single SAN **113** with a master SM selected from among the separate SMs. The invention utilizes time stamps and GUID addresses on SMDB entries as an aid for a Subnet Manager to efficiently absorb the SMDB of another subnet manager during the subnet management handover process.

Figure 8 illustrates a SAN **113** comprising two separate systems (subnets), each having its own master subnet managers, which are interlinked. System 1 **801** includes Host1 **803**, Master Subnet Manager SM1 **805**, and devices 1A **809** and 1B **807** all interconnected via subnet **810**. System 2 **811** includes Host2 **813**, Master Subnet Manager SM2 **815**, and devices 2A **819** and 2B **817** connected to subnet **820**. In the illustrative embodiment, both systems utilize the same P_Key assignments. Device A1 **809** has a P_Key assigned with a value of 1, and device 1B **807** has a P_Key assigned with a value of 5. Device 2A **819** has a P_Key assigned with a value of 2, and device 2B **817** has a P_Key assigned with a value of 5. Thus, the

two systems are joined together via a link **821** to from SAN **113**. The linking of the two systems may be completed via a switch as described above with reference to **Figure 1A**. Thus, devices 1B **807** and 2B **817** of different systems are each assigned a P_Key value of 5. In a preferred embodiment, the invention provides a method for handling the overlaps which occur in the P_Key assignments when the systems are merged as described below.

When the two systems of **Figure 8** are initially merged, one of the two master subnet managers is relegated to a standby subnet manager, while the other SM becomes the master subnet manager of the new merged subnet (SAN **113**). Selection of the particular SM that becomes the master SM of the merged system may be dependent on priority values associated with each SM during their initial configuration. In a priority scheme, the SM with the highest priority is automatically selected as the master SM. Priority value may be assigned by a subnet administrator when the subnet is initially configured. In the event that two SMs have the same priority value, then, in the preferred embodiment, the SM with the lowest GUID is selected to be the master subnet manager. During the handover process, the master subnet manager of the newly-formed merged network receives the SMDB of the other subnet manager, reconfigures the network or network components where necessary and takes over management of the total merged network.

The method utilized by the present invention to accomplish the seamless transfer of SMDB is to timestamp each SMDB with the time at which the last change was made in that SMDB. With this timestamp, the SM that is accepting the handover can then determine the correct

action to take in handling each entry as the SMDB(s) being taken over is being absorbed. Referring to **Figure 9A**, the process of time stamping entries in SMDB is illustrated. The process begins at block **901**, and following, a determination is made whether there is a change to a Subnet Management Database at block **903**. If not then the process loops back to **901**. If there was a change **903**, then the entry is time stamped with the time of the last change to the entry in that SMDB, and the process returns to block **901**.

Referring now to **Figure 9B**, the process begins at block **951**. Next, a determination is made at block **953** whether there is a merge of subnets occurring, and if no merge is occurring, the process returns to block **951**. If, however, a merge of subnets is occurring, the master subnet manager of the new subnet being formed by merging two or more other subnets begins to examine all entries of SMDBx with it's own SMDBb at block **955**. As utilized herein, SMDBx refers to the database of subnets whose subnet managers are not selected as the master subnet manager when a merger of the subnets occurs. SMDBb refers to the database of the master subnet manager prior to merger of the databases. returning to **Figure 9B**, a determination is made at block **957** whether the master subnet manager finds the same GUID entry in both SMDBx and SMBDb. If the same GUID entry is found, the master subnet manager keeps the one with the latest time stamp and discards the other at block **959**. If, however, the same GUIDs are not found, the process moves to block **961**.

Next a determination is made at block **961** whether all GUIDS have been examined in SMDBx. If not all the GUIDS have been examined, then the process returns to block **957**; however, if all the GUIDS have been examined, the process moves to block **963**, where a determination is made whether SMDBx has the same P_Key entries for different GUIDS as in SMDBb. If yes, then the master subnet manager will change all occurrences of that P_Key in SMDBx to a new unique P_Key value, which is different than that in either SMDBx or SMDBb at block **965**. The process then moves to block **967**. If the P_Key entries are not the same for SMDBx and SMDBb, then a determination is made at block **967** whether all entries in SMDBx have been examined. If all the entries have not been examined, the process returns to block **963**. If all entries have been examined in SMDBx, then the SMDBx is merged with SMDBb at block **969**. Next, a determination is made at block **971** whether there is another database to be merged with master subnet manager's SMDBb, and if so, the process returns to block **955**. Otherwise, the process ends at block **973**.

There may be other overlapping information in the respective databases that may need to be merged as described for the P_Keys and GUIDS above. These would follow a similar process as described in Figure 9B.

As a final matter, it is important to note that while an illustrative embodiment of the present invention has been, and will continue to be, described in the context of a fully functional data processing system, those skilled in

the art will appreciate that the software aspects of an illustrative embodiment of the present invention are capable of being distributed as a program product in a variety of forms, and that an illustrative embodiment of the present invention applies equally regardless of the particular type of signal bearing media used to actually carry out the distribution. Examples of signal bearing media include recordable type media such as floppy disks, hard disk drives, CD ROMs, and transmission type media such as digital and analogue communication links.

In the above detailed description of the preferred embodiments, reference is made to the accompanying drawings which form a part hereof, and in which is shown, by way of illustration, specific embodiments in which the invention may be practiced. It is to be understood that other embodiments may be utilized and structural or logical changes may be made without departing from the scope of the present invention. For example, although the invention is described with reference to multiple computing nodes accessing a single database, the invention is applicable to all other transactions occurring on the network. The above detailed description, therefore, is not to be taken in a limiting sense, and the scope of the present invention is defined by the appended claims.

CLAIMS

What is claimed is:

1 1. A method for efficiently merging subnets comprising
2 the steps of:

3
4 linking a first subnet having a first subnet manager
5 and a first database with a second subnet having a second
6 subnet manager and a second database to create a merged
7 subnet; and

8
9 dynamically selecting one of said first subnet manager
10 and said second subnet manger as a master subnet manager,
11 which controls the merged subnet.

12
1 2. The method of Claim 1, wherein said first and second
2 database comprise configuration entries including a
3 partition key (P_Key) and a global unique identifier
4 (GUID), which are time-stamped with a time said
5 configuration entries are modified by respective ones of
6 said first and second subnet managers.

1 3. The method of Claim 2, wherein said first subnet
2 manager is selected as said master subnet manager, said
3 method further comprising the step of:

4
5 absorbing said subnet configuration entries from said
6 database of the other subnet manager to said database of
7 said master subnet manager to create a merged database.

1 4. The method of Claim 3, wherein said absorbing step
2 further comprises the steps of:

3
4 determining that a first GUID entry of said second
5 database is the same as a second GUID entry of said first
6 database; and

7
8 in response to said determining step, selecting a
9 latest time-stamped GUID entry from among said first GUID
10 entry and said second GUID entry as a representative GUID
11 entry for said merged database.

1 5. The method of Claim 4, further comprising the step of
2 discarding a GUID entry not selected as said representative
3 GUID entry.

4 6. The method of Claim 4, wherein said absorbing step
5 further comprises the steps of:

6 determining that a first P_Key entry of a GUID of said
7 second database of said other subnet manager is the same as
8 a second P_key entry of a different GUID of said first
9 database;

10 in response to said determining step, changing all
11 occurrences of said P_Key in said second database to a new
P_Key value.

1 7. The method of Claim 6, wherein said changing step
2 further comprises the step of selecting only a P-Key value
3 that is not one of said P_Key values within said first
4 database and said second database.

8. The method of Claim 3, further comprising the step of configuring said merged subnet utilizing said master subnet manager.

1 9. A computer program product comprising:

2
3 a computer readable medium; and

4
5 program instructions on said computer readable medium
6 for:

7
8 linking a first subnet having a first subnet manager
9 and a first database with a second subnet having a second
10 subnet manager and a second database to create a merged
11 subnet; and

12
13 dynamically selecting one of said first subnet manager
14 and said second subnet manger as a master subnet manager,
15 which controls the merged subnet.

1 10. The computer program product of Claim 9, wherein said
2 first and second database comprise configuration entries
3 including a partition key (P_Key) and a global unique
4 identifier (GUID), which are time-stamped with a time said
5 configuration entries are modified by respective ones of
6 said first and second subnet managers.

1 11. The computer program product of Claim 10, wherein said
2 first subnet manager is selected as said master subnet
3 manager, said program product further comprising program
4 instructions for:

5
6 absorbing said subnet configuration entries from said
7 database of the other subnet manager to said database of
8 said master subnet manager to create a merged database.

1 12. The computer program product of Claim 11, wherein said
2 absorbing instructions further comprises program
3 instructions for:

4
5 determining that a first GUID entry of said second
6 database is the same as a second GUID entry of said first
7 database; and

8
9 in response to said determining step, selecting a
10 latest time-stamped GUID entry from among said first GUID
11 entry and said second GUID entry as a representative GUID
12 entry for said merged database and discarding an older
13 time-stamped entry.

14 13. The computer program product of Claim 12, further
15 comprising program instructions for discarding a GUID entry
16 not selected as said representative GUID entry.

17 14. The computer program product of Claim 12, wherein said
18 absorbing instructions further comprises program
19 instructions for:

20
21 determining that a first P_Key entry of a GUID of said
22 second database of said other subnet manager is the same as
23 a second P_key entry of a different GUID of said first
24 database;

25
26 in response to said determining step, changing all
27 occurrences of said P_Key in said second database to a new
28 P_Key value.

1 15. The computer program product of Claim 14, wherein said
2 changing instructions further comprises program
3 instructions for selecting only a P-Key value that is not
4 one of said P_Key values within said first database and
5 said second database.

1 16. The computer program product of Claim 11, further
2 comprising program instructions for configuring said merged
3 subnet utilizing said master subnet manager.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
22

1 17. A system comprising:

2
3 a first subnet with a first subnet manager and a first
4 database of subnet entries;

5
6 a second subnet with a second subnet manager and a
7 second database of subnet entries, wherein said second
8 subnet is coupled to said first subnet;

9
10 logic components for selecting a master subnet manager
11 from among said first subnet manager and said second subnet
12 manager; and

13
14 software logic associated with said master subnet
15 manager for merging said first database and said second
16 database.

1 18. The system of Claim 17, wherein:

2
3 said first subnet manager is said master subnet
4 manager; and

5
6 said first subnet manager absorbs configuration
7 entries from said second database into said first database
8 to create a merged database.

1 19. The system of claim 18, wherein said first subnet
2 manager controls and manages said system utilizing said
3 merged database.

1 20. The system of Claim 17, wherein said first subnet
2 manager determines when a first GUID entry in said first

3 database is the same as a second GUID entry in said second
4 database and dynamically selects a latest time-stamped
5 entry from among both of said first and second GUID entries
6 for inclusion in said merged database and discarding an
7 older time-stamped entry.

5 **ABSTRACT OF THE DISCLOSURE****EFFICIENT PROCESS FOR HANDOVER BETWEEN SUBNET MANAGERS**

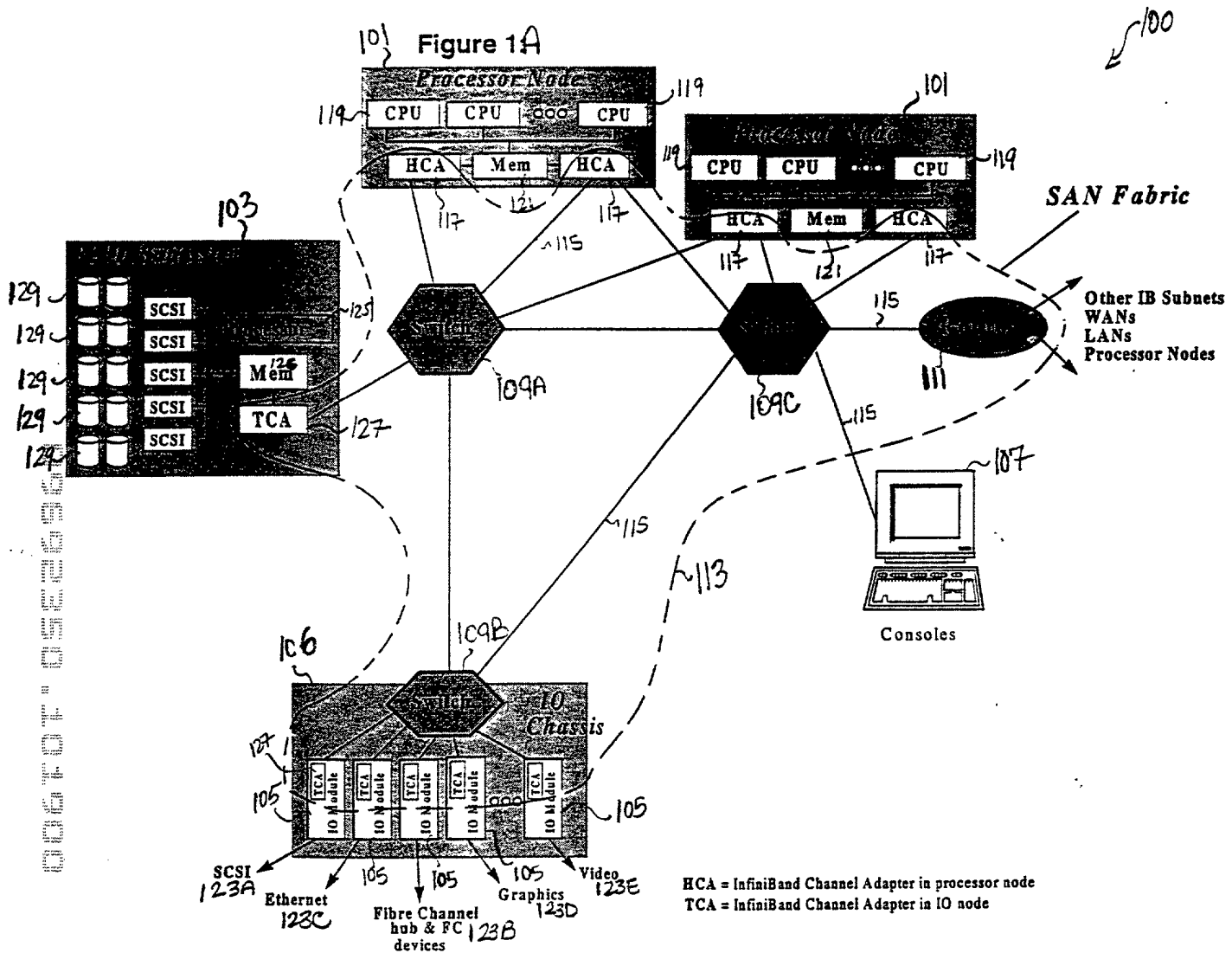
10 A method and system for efficiently merging subnets
having individual subnet managers into a single network
with one master subnet manager is disclosed. During
discovery and configuration of a subnet, a subnet manager
creates a Subnet Management Database (SMDB)
representative of the subnet components being managed.
Each subnet manager contains an independent SMDB. When
15 two or more subnets are merged (i.e., linked/connected
together) to form a single network, a single one of the
subnet managers is selected as the master subnet manager
and all the subnets SMDBs must be merged. The other
subnet managers are relegated to standby status. In a
20 preferred embodiment, a SMDB record labeling mechanism is
utilized to differentiate among components from the
different subnets that may have the same parameter
values, such as protection keys (P_keys).

25 In one embodiment of the invention, when there are two
or more separate SMDBs, each maintained by separate master
subnet managers on a separate subnet, means are provided
for efficient merging of the SMDBs, such that the separate
subnets become a single subnet, with a single master subnet
30 manager and with a single SMDB.

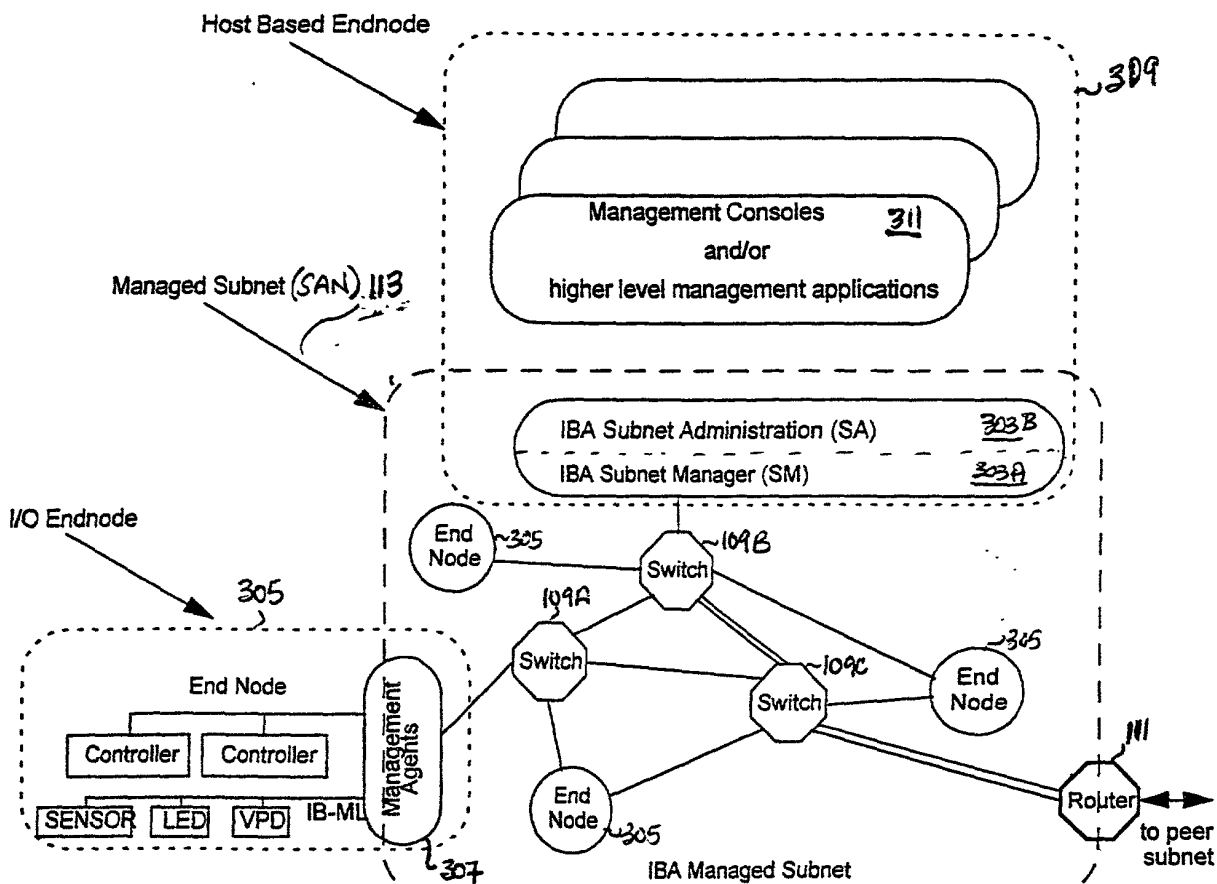
17 10

EK538216795US

AUS920000621US1

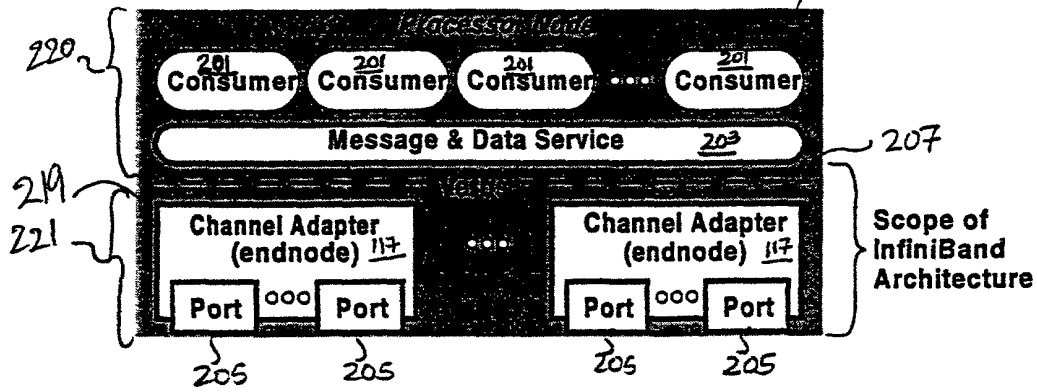


2810
FIG. 1B



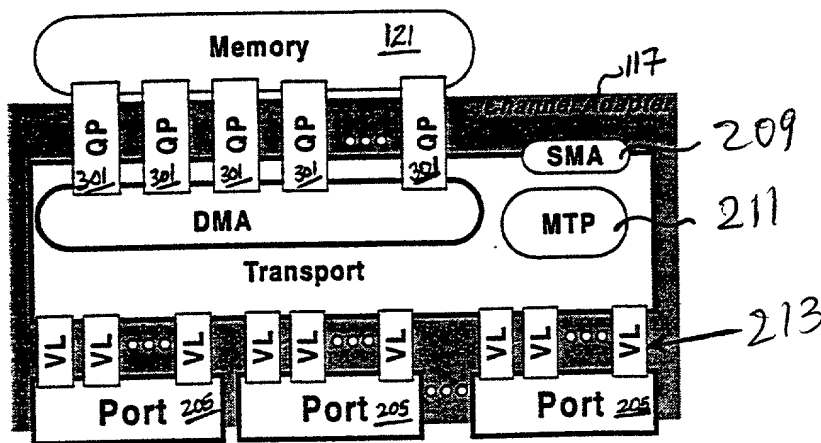
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27

101



4 of 10

Figure 3: Channel Adapter



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17

Figure 4

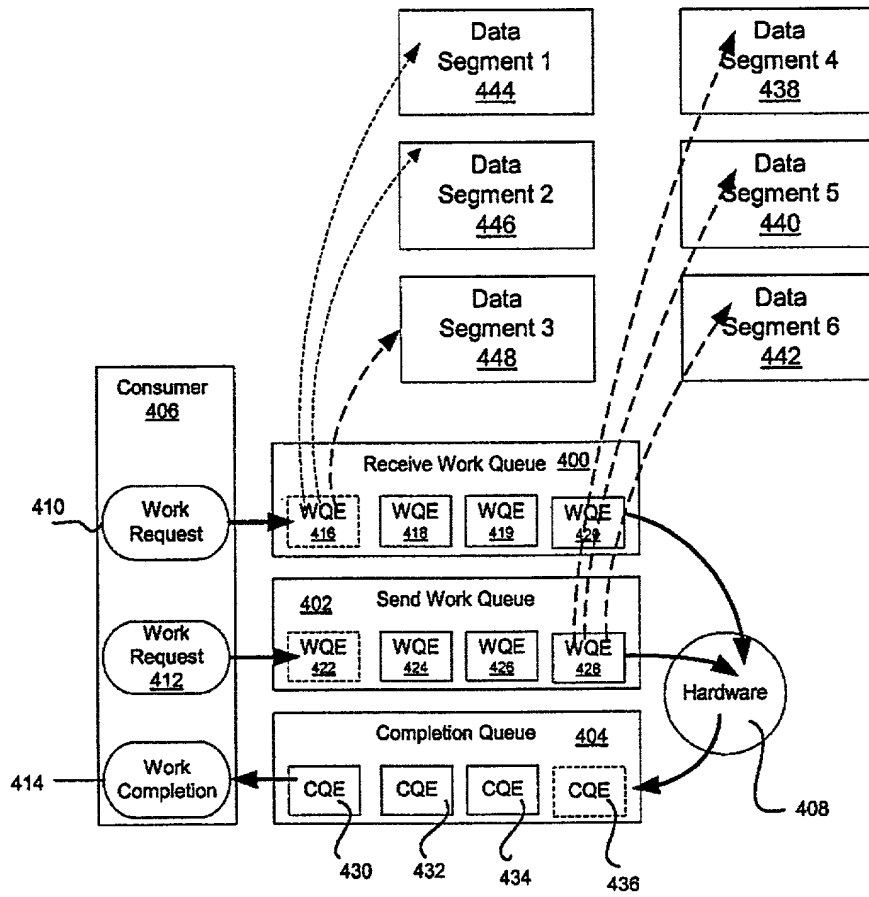
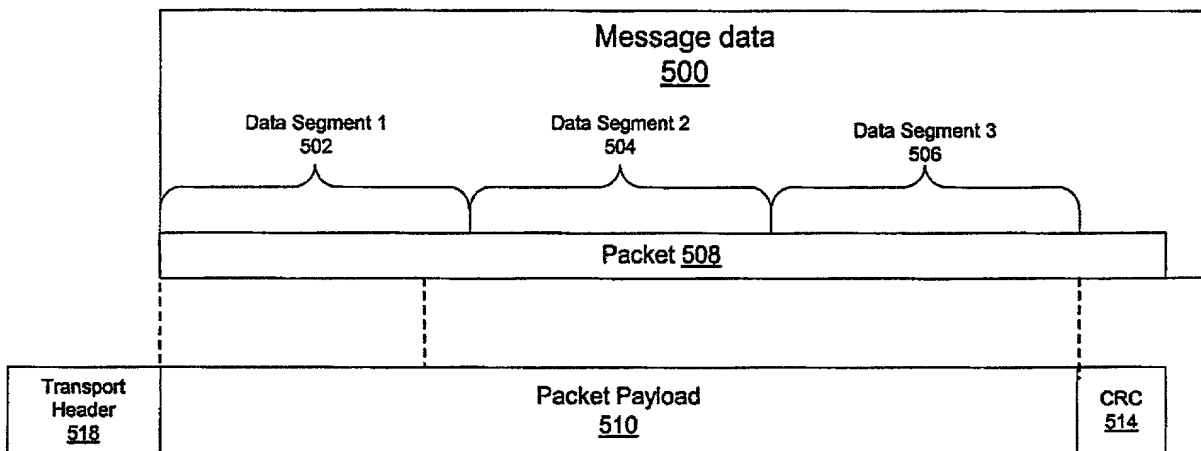
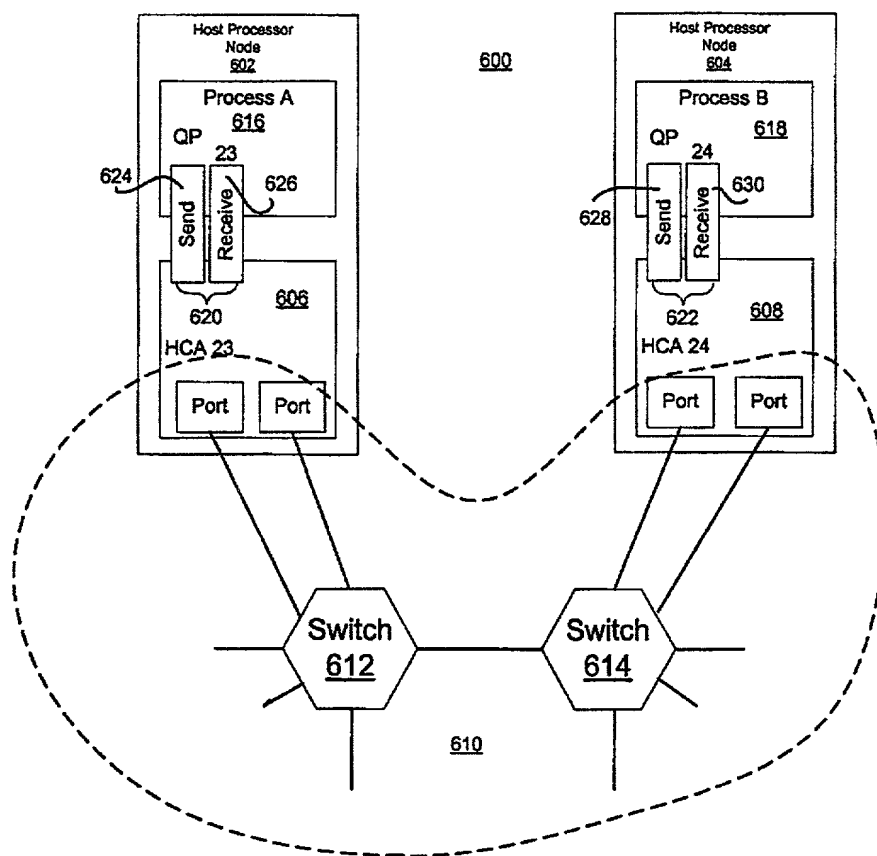


Figure 5



7810

Figure 6



8 of 10

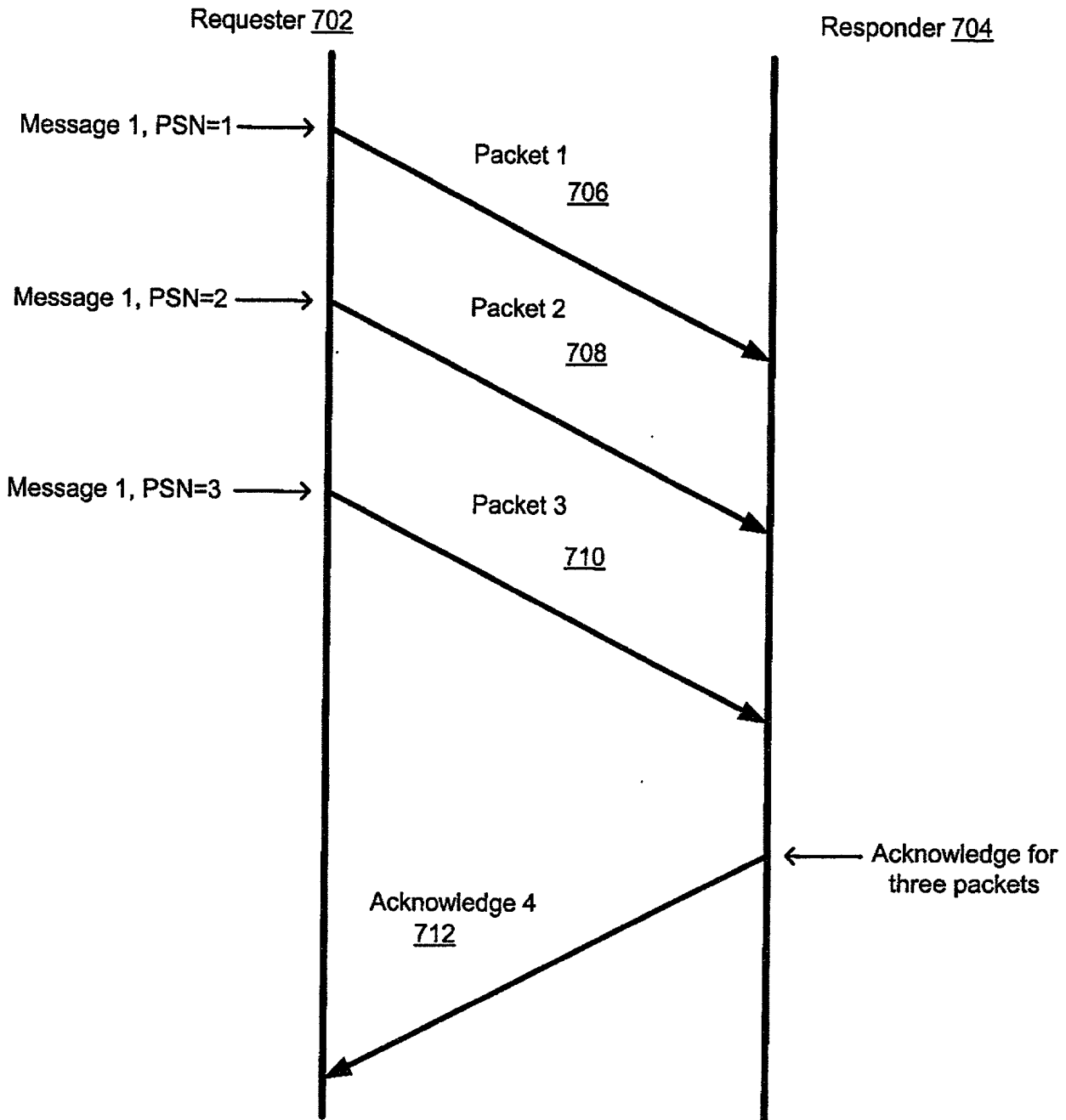


Figure 7

FIG. 8

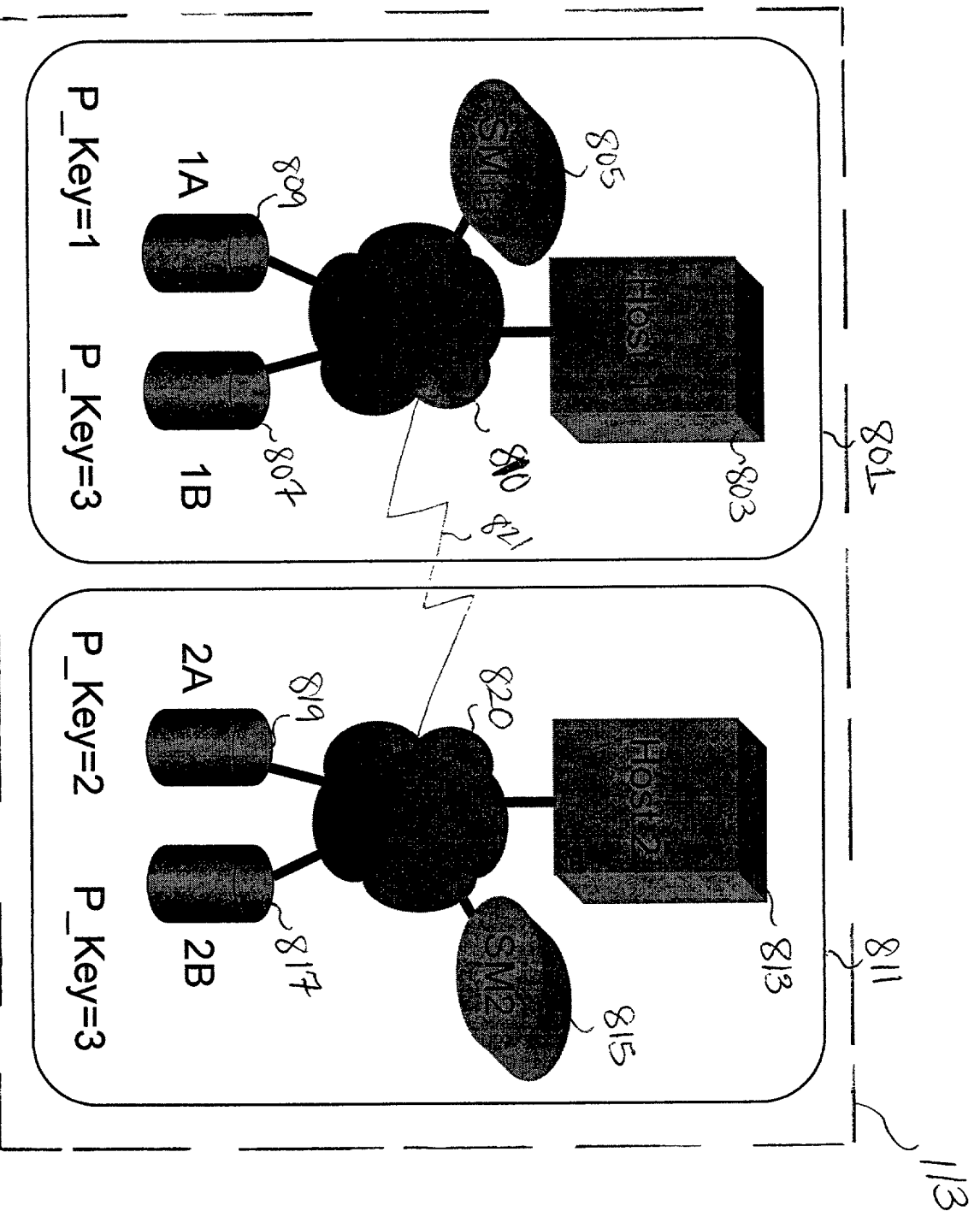
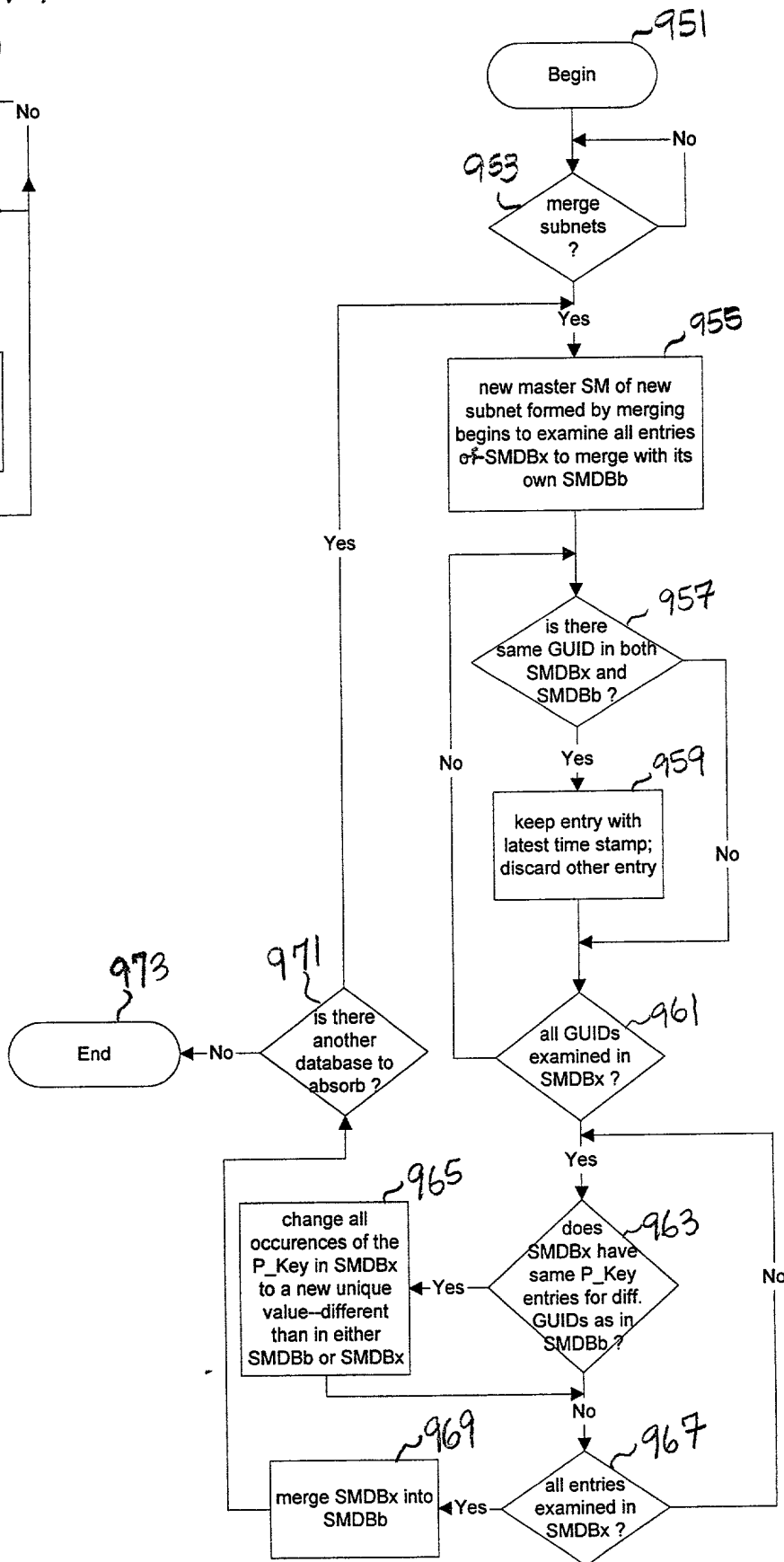
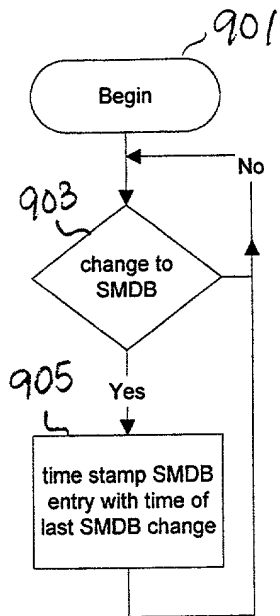


FIG. 9A

10 of 10

FIG. 9B



My residence, post office address and citizenship are as stated below next to my name;

EFFICIENT PROCESS FOR HANDOVER BETWEEN SUBNET MANAGERS

X is attached hereto.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

_____ Yes _____ No
(Number) (Country) (Day/Month/Year)

(Application Serial #)	(Filing Date)	(Status)
------------------------	---------------	----------

Page 1 of 3

DOCKET NUMBER: AUS920000621US1

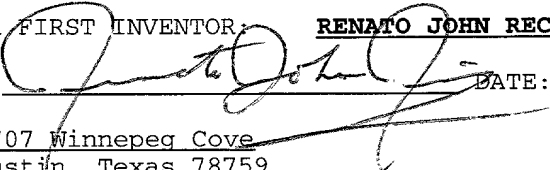
true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorneys and/or agents to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

John W. Henderson, Jr., Reg. No. 26,907; Thomas E. Tyson, Reg. No. 28,543; Robert M. Carwell, Reg. No. 28,499; Jeffrey S. LaBaw, Reg. No. 31,633; Douglas H. Lefevre, Reg. No. 26,193; Casimer K. Salys, Reg. No. 28,900; David A. Mims, Jr., Reg. No. 32,708; Volel Emile, Reg. No. 39,969; James H. Barksdale, Jr., Reg. No. 24,091; Leslie A. Van Leeuwen, Reg. No. 42,196; Marilyn S. Dawkins, Reg. No. 31,140; Mark E. McBurney, Reg. No. 33,114; Christopher A. Hughes, Reg. No. 26,914; Edward A. Pennington, Reg. No. 32,588; John E. Hoel, Reg. No. 26,279; Joseph C. Redmond, Jr., Reg. No. 18,753; Andrew J. Dillon, Reg. No. 29,634; Daniel E. Venglarik, Reg. No. 39,409; Jack V. Musgrove, Reg. No. 31,986; Brian F. Russell, Reg. No. 40,796; Matthew W. Baca, Reg. No. 42,277; Antony P. Ng, Reg. No. 43,427; Michael R. Barre, Reg. No. 44,023; Andrew Mitchell Harris, Reg. No. 42,638; Richard McCain, Reg. No. 43,785; Michael E. Noe, Jr., Reg. No. 44,975.

Send correspondence to: Andrew J. Dillon, Esq., FELSMAN, BRADLEY, VADEN, GUNTER & DILLON, LLP, Suite 350 Lakewood on the Park, 7600B North Capital of Texas Highway, Austin, Texas 78731, and direct all telephone calls to Atty. Andrew J. Dillon, (512) 343-6116.

FULL NAME OF SOLE OR FIRST INVENTOR: RENATO JOHN RECIO

INVENTORS SIGNATURE: 

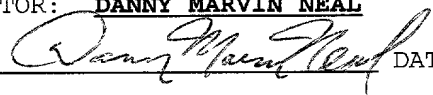
DATE: 10/16/2000

RESIDENCE: 6707 Winnepeg Cove
Austin, Texas 78759

CITIZENSHIP: U.S.A.

POST OFFICE ADDRESS: 6707 Winnepeg Cove
Austin, Texas 78759

FULL NAME OF SECOND INVENTOR: DANNY MARVIN NEAL

INVENTORS SIGNATURE: 

DATE: 10/12/2000

RESIDENCE: 4604 Hightower Drive
Round Rock, Texas 78681

CITIZENSHIP: U.S.A.

POST OFFICE ADDRESS: 4604 Hightower Drive
Round Rock, Texas 78681

DOCKET NUMBER: AUS920000621US1

FULL NAME OF THIRD INVENTOR: GREGORY FRANCIS PFISTER

INVENTORS SIGNATURE: *Gregory Francis Pfister* DATE: 10/3/00

RESIDENCE: 5905 Sir Ivor Cove
Austin, Texas 78746

CITIZENSHIP: U.S.A.

POST OFFICE ADDRESS: 5905 Sir Ivor Cove
Austin, Texas 78746

FULL NAME OF FOURTH INVENTOR: GREGORY MICHAEL NORDSTROM

INVENTORS SIGNATURE: _____ DATE: _____

RESIDENCE: 1001 Golf Course Rd.
Pine Island, Minnesota 55963

CITIZENSHIP: U.S.A.

POST OFFICE ADDRESS: 1001 Golf Course Rd.
Pine Island, Minnesota 55963

FULL NAME OF FIFTH INVENTOR: DONO VAN-MIEROP

INVENTORS SIGNATURE: _____ DATE: _____

RESIDENCE: 32 Frank Peleg Street
Haifa 34987 ISRAEL

CITIZENSHIP: THE NETHERLANDS

POST OFFICE ADDRESS: 32 Frank Peleg Street
Haifa 34987 ISRAEL

DOCKET NUMBER: AUS920000621US1
 DECLARATION AND POWER OF ATTORNEY FOR
 PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name;

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

EFFICIENT PROCESS FOR HANDOVER BETWEEN SUBNET MANAGERS

the specification of which (check one)

X is attached hereto.

_____ was filed on _____
 as Application Serial No. _____
 and was amended on _____
 (if applicable)

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s):			Priority Claimed
_____	_____	_____	Yes No
(Number)	(Country)	(Day/Month/Year)	

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose information material to the patentability of this application as defined in Title 37, Code of Federal Regulations, §1.56 which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

_____	_____	_____
(Application Serial #)	(Filing Date)	(Status)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be

Page 1 of 3

DOCKET NUMBER: AUS920000621US1

true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorneys and/or agents to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

John W. Henderson, Jr., Reg. No. 26,907; Thomas E. Tyson, Reg. No. 28,543; Robert M. Carwell, Reg. No. 28,499; Jeffrey S. LaBaw, Reg. No. 31,633; Douglas H. Lefevre, Reg. No. 26,193; Casimer K. Salys, Reg. No. 28,900; David A. Mims, Jr., Reg. No. 32,708; Vollel Emile, Reg. No. 39,969; James H. Barksdale, Jr., Reg. No. 24,091; Leslie A. Van Leeuwen, Reg. No. 42,196; Marilyn S. Dawkins, Reg. No. 31,140; Mark E. McBurney, Reg. No. 33,114; Christopher A. Hughes, Reg. No. 26,914; Edward A. Pennington, Reg. No. 32,588; John E. Hoel, Reg. No. 26,279; Joseph C. Redmond, Jr., Reg. No. 18,753; Andrew J. Dillon, Reg. No. 29,634; Daniel E. Venglarik, Reg. No. 39,409; Jack V. Musgrove, Reg. No. 31,986; Brian F. Russell, Reg. No. 40,796; Matthew W. Baca, Reg. No. 42,277; Antony P. Ng, Reg. No. 43,427; Michael R. Barre, Reg. No. 44,023; Andrew Mitchell Harris, Reg. No. 42,638; Richard McCain, Reg. No. 43,785; Michael E. Noe, Jr., Reg. No. 44,975.

Send correspondence to: Andrew J. Dillon, Esq., FELSMAN, BRADLEY, VADEN, GUNTER & DILLON, LLP, Suite 350 Lakewood on the Park, 7600B North Capital of Texas Highway, Austin, Texas 78731, and direct all telephone calls to Atty. Andrew J. Dillon, (512) 343-6116.

FULL NAME OF SOLE OR FIRST INVENTOR: RENATO JOHN RECIO

INVENTORS SIGNATURE: _____ DATE: _____

RESIDENCE: 6707 Winnepeg Cove
Austin, Texas 78759

CITIZENSHIP: U.S.A.

POST OFFICE ADDRESS: 6707 Winnepeg Cove
Austin, Texas 78759

FULL NAME OF SECOND INVENTOR: DANNY MARVIN NEAL

INVENTORS SIGNATURE: _____ DATE: _____

RESIDENCE: 4604 Hightower Drive
Round Rock, Texas 78681

CITIZENSHIP: U.S.A.

POST OFFICE ADDRESS: 4604 Hightower Drive
Round Rock, Texas 78681

DOCKET NUMBER: AUS920000621UFIFULL NAME OF THIRD INVENTOR: GREGORY FRANCIS PFISTER

INVENTORS SIGNATURE: _____ DATE: _____

RESIDENCE: 5905 Sir Ivor Cove
Austin, Texas 78746CITIZENSHIP: U.S.A.POST OFFICE ADDRESS: 5905 Sir Ivor Cove
Austin, Texas 78746FULL NAME OF FOURTH INVENTOR: GREGORY MICHAEL NORDETRON

INVENTORS SIGNATURE: _____ DATE: _____

RESIDENCE: 1001 Golf Course Rd.
Pine Island, Minnesota 55963CITIZENSHIP: U.S.A.POST OFFICE ADDRESS: 1001 Golf Course Rd.
Pine Island, Minnesota 55963FULL NAME OF FIFTH INVENTOR: DONO VAN-MIEGROFINVENTORS SIGNATURE:  DATE: 10/15/00RESIDENCE: 32 Frank Peleg Street
Haifa 34987 ISRAELCITIZENSHIP: THE NETHERLANDSPOST OFFICE ADDRESS: 32 Frank Peleg Street
Haifa 34987 ISRAEL

DOCKET NUMBER: AUS920000621US1

true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorneys and/or agents to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

John W. Henderson, Jr., Reg. No. 26,907; Thomas E. Tyson, Reg. No. 28,543; Robert M. Carwell, Reg. No. 28,499; Jeffrey S. LaBaw, Reg. No. 31,633; Douglas H. Lefevre, Reg. No. 26,193; Casimer K. Salys, Reg. No. 28,900; David A. Mims, Jr., Reg. No. 32,708; Volel Emile, Reg. No. 39,969; James H. Barksdale, Jr., Reg. No. 24,091; Leslie A. Van Leeuwen, Reg. No. 42,196; Marilyn S. Dawkins, Reg. No. 31,140; Mark E. McBurney, Reg. No. 33,114; Christopher A. Hughes, Reg. No. 26,914; Edward A. Pennington, Reg. No. 32,588; John E. Hoel, Reg. No. 26,279; Joseph C. Redmond, Jr., Reg. No. 18,753; Andrew J. Dillon, Reg. No. 29,634; Daniel E. Venglarik, Reg. No. 39,409; Jack V. Musgrove, Reg. No. 31,986; Brian F. Russell, Reg. No. 40,796; Matthew W. Baca, Reg. No. 42,277; Antony P. Ng, Reg. No. 43,427; Michael R. Barre, Reg. No. 44,023; Andrew Mitchell Harris, Reg. No. 42,638; Richard McCain, Reg. No. 43,785; Michael E. Noe, Jr., Reg. No. 44,975.

Send correspondence to: Andrew J. Dillon, Esq., FELSMAN, BRADLEY, VADEN, GUNTER & DILLON, LLP, Suite 350 Lakewood on the Park, 7600B North Capital of Texas Highway, Austin, Texas 78731, and direct all telephone calls to Atty. Andrew J. Dillon, (512) 343-6116.

FULL NAME OF SOLE OR FIRST INVENTOR: RENATO JOHN RECIO

INVENTORS SIGNATURE: _____ DATE: _____

RESIDENCE: 6707 Winnepeg Cove
Austin, Texas 78759CITIZENSHIP: U.S.A.POST OFFICE ADDRESS: 6707 Winnepeg Cove
Austin, Texas 78759FULL NAME OF SECOND INVENTOR: DANNY MARVIN NEAL

INVENTORS SIGNATURE: _____ DATE: _____

RESIDENCE: 4604 Hightower Drive
Round Rock, Texas 78681CITIZENSHIP: U.S.A.POST OFFICE ADDRESS: 4604 Hightower Drive
Round Rock, Texas 78681

